

Appendix D - Dialup Eudora

Introduction

Eudora is designed for use with MacTCP or for dialup to Cisco terminal servers. It can be used with other dialup connections as well. Complicated setups are likely to be unreliable, but if the setup commands to connect are simple, Eudora works well.

These instructions assume you are familiar with ResEdit, your communications equipment and your hosts. It is suggested that one person make these changes to Eudora and then redistribute the customized version to other users at your site.

Macintosh Requirements

To use Eudora over a dialup connection, you must have the Communications Toolbox installed. System 7 has it built in.

Dialup Requirements

Eudora needs a “transparent” connection to your POP, SMTP, and (optionally) Ph servers. Transparent means primarily two things:

1. Characters Eudora sends should NOT be echoed back to Eudora. Most systems do echo characters, so something special may need to be done to achieve this.
2. You must pay close attention to how carriage returns are treated. UNIX systems routinely translate carriage returns into linefeeds; you must either disable this on your system or teach Eudora how to deal with it.

Navigation

Eudora has a very rudimentary scripting system built in. This capability is called “navigation” to separate it from the rather sophisticated connotations of “scripting.”

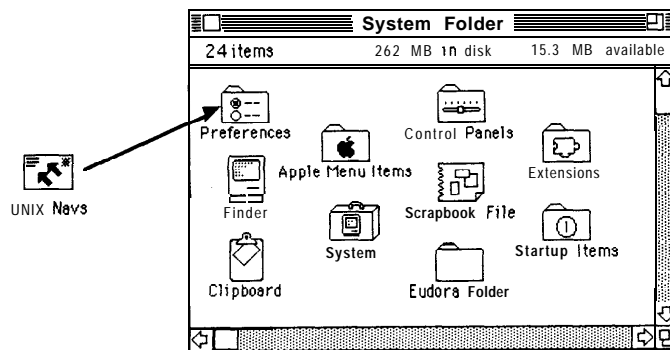
Eudora uses a list of strings which it sends out the serial port one at a time. After each string, it waits until there is no output from the remote system for 2 seconds

or until a given string is matched, after which Eudora continues with the next string.

There can be three sets of navigation strings contained in STR# resources: one is used when connecting (*Navigate In*); one is used when disconnecting (*Navigate Out*); and one is used when switching from SMTP to POP (*Navigate Mid*). You can use ResEdit to manipulate these resources. Any of the resources can be absent, in which case Eudora skips the navigation it would otherwise have done with that resource.

For each function, Eudora selects an appropriate resource in the following manner. First, it takes the name of the connection tool currently in use (e.g., “Apple Modem Tool”); then, it appends a space and the current type of navigation (e.g., “Navigate In”). Eudora looks for an STR# resource with that name (e.g., “Apple Modem Tool Navigate In”). If that is not found, Eudora looks for an STR# resource named after the navigation type (e.g., “Navigate In”). Finally, it looks for specific resource ids (2400 for Navigate In, 2600 for Navigate Out, and 4200 for Navigate Mid). The first STR# resource found is the only one used.

Navigation resources can be in the Eudora application itself, in your Eudora Settings file, or in a *plug-in* file. The latter is a file with type “rsrc” and creator “CSOm” placed in your Preferences folder; use of a plug-in file is highly recommended.



Plug-In files go in the Preferences folder.

Special Sequences in Navigation Resources

Eudora provides a set of special character sequences for use in Navigation resources. These sequences are all two characters long, the first character being a backslash. They are either replaced with items from your Eudora Settings, or they modify the Navigation process.

Replacements

These special characters are replaced with strings. They can appear at any place in a navigation string.

\u POP account user name.

\h POP account host name.

\p POP account password.

\s SMTP server host name.

\U Dialin user name.

\P Dialin password.

\n A linefeed (ASCII 10).

\r A carriage return (ASCII 13).

\\ A single backslash character.

^\nEnter a prompt for the \a modifier.

\a Ask the user for something.

\c Cancel if a string appears.

When ^ appears, all the text after it (up until the end of the string or the next '\') goes into a prompt string, that will be used with the \a modifier.

When \a appears, a dialog will be used to ask the user to type something. What the user types will be used where the '\a' appears.

Example: ^\nEnter the SecureId code below:\a\r

This will put up a dialog with the prompt "Enter the SecureId code below:", and then wait for the user to type in the code. The code will then be sent, along with a carriage return.

\c is used to enter text that will abort the navigation process. The text is valid ONLY for the duration of the particular string it appears in. If you wish to look for it in another string, you must specify it again.

Example: \P\r\accepted\cdenied

Eudora will send the dialin password and a return, then wait until it sees either “accepted” (in which case the navigation process will continue) or “denied” (which will abort the navigation process and the mail transfer).

Modifiers

\b Hide from Progress window.

\D Delay.

\B Break.

\e Expect something.

\b – when this appears as the first character in a given a string, Eudora won’t print the string in the Progress window when it is sent. This is useful for passwords or other state secrets.

\D – should be the first character in the string and followed only by digits or an expect string. The digits are taken as a number of seconds to delay (e.g., “\D2” is a two-second delay). As with other strings, Eudora will wait for output to stop for two seconds, or for an expect string, before proceeding.

\B – should be the first character in the string and followed only by digits or an expect string. The digits are taken as the number of ticks (60ths of a second) to send a break signal (e.g., “\B30” is a half-second break). As with other strings, Eudora waits for output to stop for two seconds, or for an expect string, before proceeding.

\e – makes Eudora expect to see a specific string in the output from the dialup server. Eudora sends whatever comes before the \e, and then waits for whatever comes after it to occur in the data sent from the host. There are two caveats to this. First, Eudora only matches on the first 7 bits; the high bit of each character is ignored for matching. This is needed for systems that use parity. Second, Eudora’s matching is fast and sloppy; highly repetitive data streams and long expect strings might fool it (for example, Eudora wouldn’t see “Login:” if your server said: “LogLogin:”).

Eudora follows its normal time-out process when looking for an expect string. That is, after 45 seconds Eudora asks you if you want to keep waiting or cancel the process. If you cancel, the connection process is stopped.

If you use either of the password replacement sequences (“\p” for your POP account password or “\P” for your dialup password) in the same string as an expect, Eudora assumes the password is wrong if the expect string isn’t found. This causes Eudora to ask for your password the next time it tries to connect.

Connecting to Servers

Once the navigation is done, Eudora issues a command to connect to the proper port of the server you are using. Once this command is sent, the connection **MUST** be transparent, as discussed above; no echoes and no carriage return translation. Eudora comes configured to send the command:

```
telnet hostname portnumber /stream<return>
```

The “<return>” means a carriage return in this document only; you must type actual carriage returns in ResEdit. A template for the command is kept in 7400.13 (“7400.13” is shorthand for STR# resource id 7400, string 13). The template begins life as “telnet %p %d /stream\n”. The %p is replaced with the hostname and the %d with the port number. It is acceptable to change this string however you please, except that %d, if it is used, must come after %p. The best way to change this string is not to modify it, but to override it. You can do this by creating an ‘STR ‘ resource of id 7413 and putting the string you want Eudora to use in that.

A Return By Any Other Name

When Eudora is communicating with your POP or SMTP server, it’s important that they agree on what constitutes a line. The Internet specification stipulates that a line ends with a carriage return followed by a linefeed. Most UNIX systems will “helpfully” translate carriage returns into linefeeds. If you can’t get that feature turned off, it may help to make Eudora send only a carriage return, and not the carriage return/linefeed pair. You can edit 6000.17 to be what you want Eudora to send at the end of a line. The best way to change this string is not to modify it, but to override it. You can do this by creating an ‘STR ‘ resource of id 6017 and putting the string you want Eudora to use in that.

An Example: Direct Connection To A Unix Machine

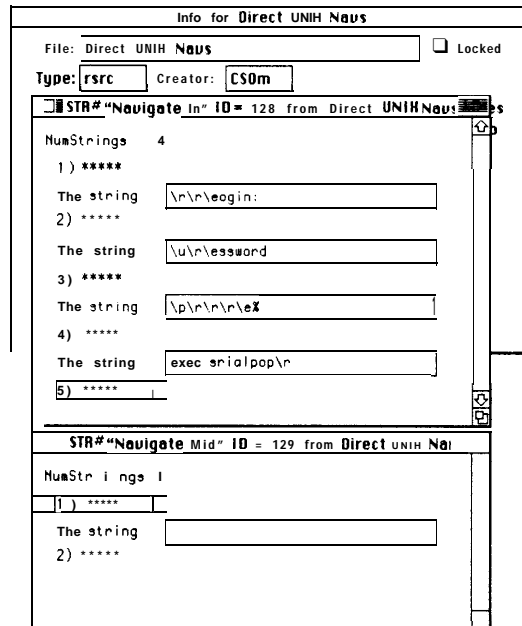
Here’s a suggested connection method if your Macintosh has a serial line to a UNIX machine, or if there are modems on your Macintosh and UNIX machine.

1. Install the srialpop program on your UNIX system. Source to this is part of the Eudora distribution.
2. Put the following strings in the “Navigate In” resource.
 - 1: \r\r\rlogin:
 - 2: \u\r\eword:

3: \p\r\r\r\re%

4: *exec srialpop*

- Put one empty string in the “Navigate Mid” resource.



Resources for dialing directly into a UNIX machine.

Srialpop takes care of the terminal settings for you.

Note: A *plug-in* with these strings in it is part of the *Eudora* distribution.
The file name is “Direct UNIX Navs.”

Need More Options?

If Eudora's scripting isn't adequate for you, one alternative is the Calypso connection tool. It lets you use CCL scripts for the connection/disconnection process. Calypso can be found on major Macintosh archive sites.